

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-214201

(43)Date of publication of application : 11.08.1998

(51)Int.Cl.

G06F 11/28

G06F 9/06

G06F 15/78

(21)Application number : 09-015743

(71)Applicant : MITSUBISHI ELECTRIC
CORP
MITSUBISHI DENKI SYST
LSI DESIGN KK
INTERNATL BUSINESS
MACH CORP <IBM>

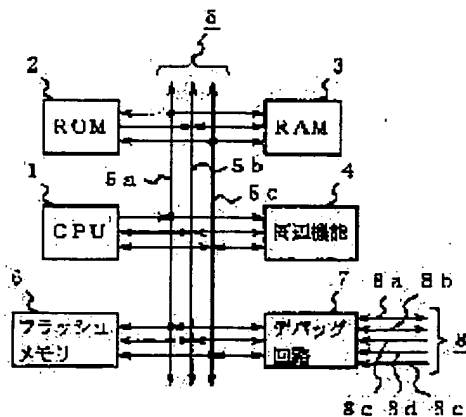
(22)Date of filing : 29.01.1997

(72)Inventor : ITO SAKAE
KANZAKI TERUAKI
AKATSUKI TADAYUKI
SAKAI TATSUYA
NUMATA TSUTOMU
NAKAMURA YASUHIRO

(54) MICROCOMPUTER

(57)Abstract:

PROBLEM TO BE SOLVED: To obtain a microcomputer in which a debug circuit equipped with various functions used at the time of program debugging is provided in the microcomputer, and a flash memory electrically capable of writing/erasing incorporated on the same chip can be used as an emulation memory. SOLUTION: This device is provided with a flash memory 6 electrically capable of writing/erasing in which a program in a development stage is stored and a debug circuit 7 having an exclusive input and output terminal 8 for connection with an outside ICE are incorporated in this device. The debug circuit 7 is provided with a communicating function with a CPU 1, a communicating function with the ICE, the trace function of the operating state of the CPU 1, break function for generating debug interruption,



function for writing a program code from the ICE in the flash memory 6, and function for transmitting the content of the flash memory 6 to the ICE.

LEGAL STATUS

[Date of request for examination] 03.04.2003

[Date of sending the examiner's decision of rejection] 06.07.2004

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-214201

(43) 公開日 平成10年(1998) 8月11日

(51) Int. CL ⁶	識別記号	P I	L
G 0 6 F 11/28		G 0 6 F 11/28	
9/06	5 4 0	9/06	5 4 0 L
15/78	5 1 0	15/78	5 1 0 K

審査請求 未請求 請求項の数 4 O L (全 21 頁)

(21) 出願番号 特願平9-15743

(22) 出願日 平成9年(1997) 1月29日

(71) 出願人 000006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(71) 出願人 391024515

三菱電機システムエル・エス・アイ・デザ
イン株式会社

兵庫県伊丹市中央3丁目1番17号

(74) 代理人 弁理士 田澤 博昭 (外1名)

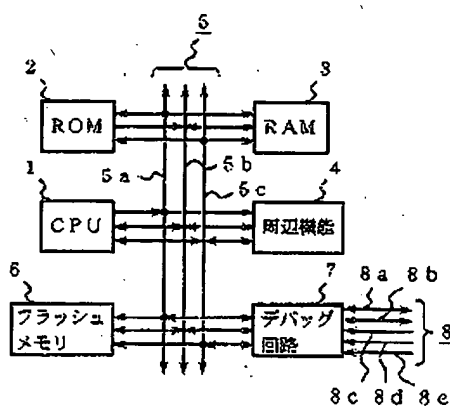
最終頁に続く

(54) 【発明の名称】 マイクロコンピュータ

(57) 【要約】

【課題】 マイコンにデバッグ専用のモードを持たせる必要があり、エミュレーションメモリを内蔵ROMと同等の速度でアクセスしようとした場合、高速動作が可能な高価なメモリが要求される。

【解決手段】 開発段階のプログラムが格納される電気的に書き込み/消去が可能なフラッシュメモリ6と、外部のICE14との接続のための専用入出力端子8を持つデバッグ回路7とを内蔵し、デバッグ回路7にCPU1との通信機能、ICE14との通信機能、CPU1の動作状態のトレース機能、デバッグ割り込みを発生させるブレイク機能、ICE14からのプログラムコードをフラッシュメモリ6に書き込む機能、フラッシュメモリ6の内容をICE14に送る機能を持たせた。



5 : 内部バス

8 : 専用入出力端子

8 a : G Dバス (データ端子)

8 b : G C L K * 端子 (クロック端子)

8 c : G O E * 端子 (制御端子)

8 d : V p p 端子 (制御端子)

8 e : R E S E T * 端子 (制御端子)

(2)

特開平10-214201

1

【特許請求の範囲】

【請求項1】 内部バスを介して接続された読み取り専用記憶装置に格納されているプログラムを用いて動作する中央演算処理装置を備えたマイクロコンピュータにおいて、

前記読み取り専用記憶装置に格納されるプログラムの開発段階のプログラムが格納される、電気的に書き込み／消去が可能なフラッシュメモリと、

前記内部バスを介して前記中央演算処理装置に接続されるとともに、外部のエミュレータと接続するための専用入出力端子を持ったデバッグ回路を設け、前記デバッグ回路に、

前記内部バスを介した前記中央演算処理装置との通信機能、

前記専用入出力端子を介した前記エミュレータとの通信機能、

前記中央演算処理装置の動作状態を示す情報を前記専用入出力端子から前記エミュレータに送るトレース機能、

前記中央演算処理装置の動作状態が、前記中央演算処理装置あるいはエミュレータがあらかじめ設定した条件に一致した時にデバッグ専用の割り込みを発生させるブレーク機能、および前記エミュレータから前記専用入出力端子を介して送られてきたプログラムコードを前記フラッシュメモリに書き込むとともに、前記フラッシュメモリの内容を読み出して前記専用入出力端子から前記エミュレータに送る機能を持たせたことを特徴とするマイクロコンピュータ。

【請求項2】 内蔵されたデバッグ回路を外部のエミュレータと接続するための専用入出力端子を、

前記デバッグ回路と前記エミュレータの間でデータを双方向にやりとりするデータ端子と、

前記データ端子によってデータをやりとりする時に、その同期をとるためのクロックを双方向に送るクロック端子と、

前記エミュレータから前記デバッグ回路に入力され、前記データ端子およびクロック端子の信号の伝搬方向を決める制御端子とによって構成したことを特徴とする請求項1記載のマイクロコンピュータ。

【請求項3】 内蔵されたデバッグ回路が、中央演算処理装置によるプログラムコードの実行アドレスがあらかじめ設定されたアドレスに一致した時にデバッグ専用の割り込みを発生させる機能を有し、

書き換え不可能な読み取り専用記憶装置に格納されたプログラムのプログラムコードにバグがある場合に、前記デバッグ回路が発生するデバッグ専用の割り込みによって、前記プログラムのプログラムコードにバグのあるプログラム部分を回避し、別途用意された前記バグのあるプログラム部分に相当する正しいプログラム部分に置き換えることを特徴とする請求項1または請求項2記載のマイクロコンピュータ。

2

【請求項4】 内蔵されたデバッグ回路が、中央演算処理装置によるプログラムコードの実行アドレスがあらかじめ設定されたアドレス範囲外となった時にデバッグ専用の割り込みを発生させる機能を有し、

前記中央演算処理装置によるプログラム実行が想定されたアドレス範囲の外に及んだ場合に、前記デバッグ回路が発生するデバッグ専用の割り込みによって、最速の検出を行うことを特徴とする請求項1または請求項2記載のマイクロコンピュータ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、中央演算処理装置（以下、CPUという）が用いるプログラムが格納される読み取り専用記憶装置（以下、ROMという）を内蔵したマイクロコンピュータ（以下、マイコンという）に関するものであり、特に、そのROMに格納するプログラムの開発を容易にするマイコンに関するものである。

【0002】

【従来の技術】図21は従来のメモリ内蔵型のマイコンの構成を示すブロック図である。図において、1は当該マイコンのCPUであり、2はそれが用いるプログラムが格納されるROMである。3は読み出し／書き込み可能な随時書き換え記憶装置（以下、RAMという）であり、4はこのマイコンの周辺機能である。5はこれら各部を接続する内部バスであり、5aはアドレスバス、5bはデータバス、5cは制御バスである。

【0003】次に動作について説明する。CPU1は当該マイコンを動作させるためのプログラムをROM2より内部バス5を介して順次読み込み、読み込んだプログラムコードを逐次解釈、実行する。このCPU1でのプログラムの実行にともなって、CPU1は内部バス5を介してRAM3あるいは周辺機能4とデータのやり取りを行い、種々のデータ処理を実行する。

【0004】なお、ROM2は読み出し専用で、このROM2へのプログラムの書き込みは半導体の製造工程中に行われるものであって、それ以降のプログラムの書き換えはできない。したがって、ユーザが開発段階でプログラムの作成およびデバッグを行う場合には、以下のような手法がとられる。

【0005】ここで、図22はマイコンのROM2に格納するプログラムを開発する際のプログラム開発環境を示すブロック図であり、相当部分には図21と同一符号を付してその説明を省略する。図において、11は図21にその構成を示したマイコンであり、12はマイコン11の内部バス5を外部に引き出した外部バス、13はこの外部バス12によってマイコン11に接続されたエミュレーションメモリであり、このエミュレーションメモリ13はマイコン11とともにボットと呼ばれる部分に格納されている。14はエミュレーションメモリ13

(3)

特開平10-214201

3

をマイコン11に接続している外部バス12につながれたエミュレータ（以下、ICEという）であり、15はこのICE14に接続された、パーソナルコンピュータ（以下、パソコンという）あるいはエンジニアリング・ワーキング・ステーション（以下、EWSという）などによるホストコンピュータである。16はユーザ基板であり、このユーザ基板16上のマイコン11を載せるスペースに設置されたソケットには、ボッドよりケーブルによって引き出されたマイコン11の各端子が接続されている。

【0006】なお、マイコン11はプログラムの開発時だけに使用する特殊なモードを備えており、そのモードでは内蔵しているROM2のアクセスを禁止するとともに、内部バス5を外部に引き出す機能を用意する必要がある。その外部に引き出された外部バス12に、読み出し／書き込みが可能なメモリ（通常スタティックRAMが使用される）によるエミュレーションメモリ13を、ROM2のエミュレーションのために接続する。すなわち、ROM2の代わりにマイコン11の外部に、外部バス12を介して読み出し／書き込みが可能なエミュレーションメモリ13を用意し、このエミュレーションメモリ13上に開発中のプログラムをマイコン11のCPU1に実行させ、バグがあれば順次それを修正してゆくことによりプログラムを完成させるものである。

【0007】ユーザは通常ホストコンピュータ15上で、C言語あるいはアセンブラ言語等によってプログラムを記述する。プログラムの記述が終わると、これをコンパイルして16進フォーマットのファイルにし、それをICE14に転送する。ICE14はマイコンとエミュレーションメモリ13を接続している外部バス12につながっており、この外部バス12を介してホストコンピュータ15から送られてきたプログラムコードをエミュレーションメモリ13に転送（ダウンロード）する。またICE14はこの外部バス12以外にも、デバッグ時にマイコン11の動作を制御するいくつかの制御信号（リセット入力、デバッグ割り込み要求入力等）、およびマイコン11の動作状況を外部からモニタするためのいくつかのモニタ信号で、マイコン11と接続されている。ICE14はエミュレーションメモリ13にプログラムをダウンロードすると、マイコン11をリセットスタートさせる。

【0008】これにより、マイコン11はエミュレーションメモリ13上のプログラムの実行を開始する。プログラム実行中の外部バス12の信号およびモニタ信号はICE14内部のトレースメモリに一時的に蓄えられる。このトレースメモリの内容はリアルタイムにホストコンピュータ15側から見るができるようになっており、ユーザは自分の作成したプログラムの実行状態をチェックすることができる。またICE14には、外部バス12の信号およびモニタ信号を随時監視して、マイ

4

コン11の動作状態があらかじめ設定した条件に一致した場合に、デバッグ専用の割り込みをかけてプログラムの実行を中断させたり、中断した状態でその時のマイコン11の内部レジスタやメモリの内容を見たり変更したりする機能を備えている。ユーザはこのような機能を駆使して、作成したプログラムが意図通りに動作していない場合にはその原因を探り、プログラムにミスがあればホストコンピュータ15上でこれを修正し、再度ICE14を介してエミュレーションメモリ13へダウンロードする。このような操作を繰り返しながらプログラムの開発作業が進められる。

【0009】

【発明が解決しようとする課題】従来のマイクロコンピュータは以上のように構成されているので、内蔵するROM2のアクセスを禁止し、エミュレーションメモリ13を外付けするために内部バス5を外部に出力する機能や、デバッグ専用の割り込み入力を許可したり、ICE14がマイコン11内部の動作状態を知るためのモニタ信号を出力する機能を持ったデバッグ時専用の特殊なモードを持つ必要があり、場合によっては、ROM2を内蔵しないプログラム開発用途専用のチップを準備しなければならない場合もあるなどの課題があった。

【0010】また、ROM2はチップ内に内蔵されているため高速でアクセスすることができ、これと同等の速度で外部のエミュレーションメモリ13をアクセスしようとした場合、高価で高速なメモリが必要となり、半導体プロセスの進歩でチップ内部の動作は年々早くなっているため、内蔵されたROM2と同じ速度で外部に接続したエミュレーションメモリ13をアクセスすること自体が難しくなりつつあるばかりか、ICE14はマイコン11の内部バス5の信号とモニタ信号を監視して、あらかじめ設定された状態を検出した場合に、マイコン11に対してデバッグ割り込みを要求してプログラムの実行を中断させる機能（ブレイク機能）を持つが、状態検出をしてから割り込み要求までに時間遅延が生じ、割り込みが受け付けられて実際にプログラムの実行が中断された時には、最初の設定よりプログラムの実行が先に進んでしまっていることが多く、さらには、マイコン11の各端子はボッドからケーブルで引き出されてユーザ基板16のソケットなどに接続されているため、そのケーブル等での信号遅延の影響で、入出力信号のタイミングがマイコン11をユーザ基板16に直接取り付け付けた場合と等価にならないことがあるなどの課題もあった。

【0011】この発明は上記のような課題を解決するためになされたもので、マイコン内部にプログラムデバッグ時に使用する種々の機能を備えたデバッグ回路を持ち、かつ同一チップ上に内蔵した電気的に書き込み／消去が可能なフラッシュメモリをエミュレーションメモリとして使用することのできるマイクロコンピュータを得ることを目的とする。

(4)

特開平10-214201

5

6

【0012】

【課題を解決するための手段】請求項1記載の発明に係るマイクロコンピュータは、開発段階のプログラムが格納される電気的に書き込み/消去が可能なフラッシュメモリと、外部のICEと接続するための専用入出力端子を備えて、CPUとの通信機能、ICEとの通信機能、CPUの動作状態のトレース機能、デバッグ専用の割り込みを発生させるブ레이크機能、ICEからのプログラムコードをフラッシュメモリに書き込む機能、フラッシュメモリの内容をICEに送る機能を有するデバック回路を内蔵させたものである。

【0013】請求項2記載の発明に係るマイクロコンピュータは、内蔵されたデバック回路を外部のICEと接続するための専用入出力端子を、データを双方向にやりとりするデータ端子、やり取りされるデータの同期をとるためのクロックを双方向に送るクロック端子、およびデータ端子とクロック端子の信号の伝搬方向を決める制御端子によって構成したものである。

【0014】請求項3記載の発明に係るマイクロコンピュータは、書き換え不可能なROMに格納されたプログラムにバグがあるとき、実行アドレスがバグのあるプログラム部分のアドレスになるとデバック回路よりデバック割り込みを発生させ、そのバグのあるプログラム部分を回避して、別途用意された正しいプログラム部分に置き換えるようにしたものである。

【0015】請求項4記載の発明に係るマイクロコンピュータは、CPUのプログラム実行が想定されたアドレス範囲外に及ぶとデバック回路よりデバック割り込みを発生させ、暴走検出を行うようにしたものである。

【0016】

【発明の実施の形態】以下、この発明の実施の一形態を説明する。

実施の形態1. 図1はこの発明の実施の形態1によるメモリ内蔵型のマイコンの構成を示すブロック図である。図において、1は当該マイコンの全体制御を掌するCPUであり、2はこのCPU1を動作させるためのプログラムが格納されるROMである。3はCPU1がROM2に格納されたプログラムを実行する際に用いるRAMであり、4はタイマ、アナログ/デジタル変換器、シリアル入出力回路などによるこのマイコンの周辺機能である。5はこれらCPU1、ROM2、RAM3、および周辺機能4を接続している内部バスであり、5aはこの内部バス5を形成するアドレスバス、5bは同じくデータバス、5cは同じく制御バスである。なお、これらの各部は図21に同一符号を付して示した従来のマイコンにおけるそれらと同等のものである。

【0017】また、6はROM2に格納されるプログラムの開発段階のプログラムが格納され、当該プログラム開発時のシミュレーションメモリとして使用される、電気的に書き込みおよび消去することが可能なフラッシュ

メモリである。7は内部バス5を介してCPU1に接続され、CPU1との通信機能、当該マイコンの外部に接続されるICEとの通信機能、CPU1の動作状態のトレース機能、デバッグ専用の割り込みを発生させるブ레이크機能、フラッシュメモリ6のリード/プログラム/イレースなどを行うダウンロード機能を有するデバック回路である。8はこのデバック回路7と外部のICEとを接続するための専用入出力端子であり、データを双方向にやりとりするデータ端子としてのGDバス8a (GD7~GD0)、やり取りされるデータの同期をとるためのクロックを双方向に送るクロック端子としてのGCLK* 端子8b、およびGDバス8aとGCLK* 端子8bの信号の伝搬方向を決めるGOE* 端子8cや、Vpp端子8d、RESET* 端子8eなどの制御端子にて構成されている。

【0018】また、図2はマイコンのROM2に格納するプログラムを開発する際のプログラム開発環境を示すブロック図である。図において、11は図1にその構成を示したマイコンであり、14はこのマイコン11の外部に接続されたICE、15はこのICE14に接続された、パソコンあるいはEWSなどによるホストコンピュータである。16はユーザ基板であり、このユーザ基板16にはマイコン11が直接搭載されている。17はユーザ基板16に設けられ、マイコン11の専用入出力端子8とICE14を接続するためのケーブルとを結合するためのコネクタである。

【0019】さらに、図3はそのマイコン11に内蔵された上記デバック回路7の構成を示すブロック図である。図において、21はデバック制御レジスタ(0)、22はデバック制御レジスタ(1)であり、これらデバック制御レジスタ(0)21およびデバック制御レジスタ(1)22は内部バス5のアドレスバス5a (A19~A0)、データバス5b (D15~D0)、および制御バス5cによってマイコン11のCPU1と接続され、その内容をCPU1から読み出し/書き込みができるようになっている。また、これらデバック制御レジスタ(0)21とデバック制御レジスタ(1)22は、GDバス8aおよびGCLK* 端子8bを介して、その内容をマイコン11の外部に接続したICE14から読み出し/書き込みすることもできる。なお、このデバック制御レジスタ(0)21およびデバック制御レジスタ(1)22はともに8ビット長のレジスタによって形成されており、それぞれの各ビットの内容によって、後述するブ레이크/トレース/ダウンロードの各機能を選択したり、それら各機能の動作条件を設定したりすることができる。

【0020】これらデバック制御レジスタ(0)21およびデバック制御レジスタ(1)22のビット構成を図4に示す。図4(a)にはデバック制御レジスタ(0)21のビット構成が示されており、その第6ビットはト

(5)

特開平10-214201

7

8

レースモードの禁止/許可を選択するためのビット、第5ビットはブレークモードの禁止/許可を選択するためのビット、第4ビット～第3ビットはオペコードアドレス、データ読み出し、データ書き込み、データ読み出し・書き込みなどのイベント検出サイクルを選択するためのビットであり、第2ビット～第0ビットはアドレス・データ一致検出、アドレス一致検出、アドレス範囲検出などのブレーク検出条件を選択するためのビットである。また、図4(b)にはデバッグ制御レジスタ(1)22のビット構成が示されており、その第4ビットはデ

10 バック割り込み要求のためのビット、第3ビットはサービスエンドフラグのためのビット、第2ビットはサービスリクエストフラグのためのビット、第1ビットはビジー(BUSY)フラグのためのビットであり、第0ビットはダウンロードの禁止/許可を選択するためのビットである。
【0021】また、図3において、23はアドレス比較レジスタ(0)、24はアドレス比較レジスタ(1)であり、これらアドレス比較レジスタ(0)23およびアドレス比較レジスタ(1)24は、ブレークモード時に

20 ブレーク検出を行う際のアドレス条件を設定する20ビット長のレジスタにて形成されている。ただし、アドレス比較レジスタ(0)23は後述するデータ比較レジスタの内容をビット単位でマスクするマスクレジスタとしての機能も持っている。このアドレス比較レジスタ(0)23とアドレス比較レジスタ(1)24は、デバッグ制御レジスタ(0)21およびデバッグ制御レジスタ(1)22と同様に、CPU1およびICE14の両者からその内容を読み出し/書き込みすることができるようになっている。

【0022】25は前記データ比較レジスタであり、ブレークモード時にブレーク検出を行う際のデータ条件を設定する16ビット長のレジスタにて形成されている。このデータ比較レジスタ25もまた、CPU1およびICE14の両者からその内容を読み出し/書き込みが行えるようになっている。

【0023】26は各々が8ビット長の8本のレジスタで形成されたデバッグレジスタであり、それぞれデバッグレジスタ(0)、デバッグレジスタ(1)、...

デバッグレジスタ(7)と命名されている。このデバッグレジスタ26はプログラムのデバッグを行う際に、CPU1とICE14との間でデータのやり取りをする時のバッファレジスタの役割を担っている。なお、このデバッグレジスタ26もまた、CPU1およびICE14の両者からその内容を読み出し/書き込みが行えるようになっている。

【0024】27はアドレス一致比較回路(0)であり、このアドレス一致比較回路(0)27はアドレス比較レジスタ(0)23の内容とCPU1からアドレスバス5aに出力されるアドレス信号(A19～A0)を随

時比較し、両者の内容が一致する場合にAOEQ信号を“H”レベルに、また後者が前者より大きい場合にAOGT信号を“H”レベルにするものである。なお、これらのAOEQ信号およびAOGT信号は後述するイベント検出回路に送られる。

【0025】28はアドレス一致比較回路(1)であり、このアドレス一致比較回路(1)28はアドレス比較レジスタ(1)24の内容とCPU1からアドレスバス5aに出力されるアドレス信号(A19～A0)を随

20 時比較し、両者の内容が一致する場合にA1EQ信号を“H”レベルに、また後者が前者より小さい場合にA1LT信号を“H”レベルにするものである。なお、これらのA1EQ信号およびA1LT信号も、後述するイベント検出回路に送られる。
【0026】29はデータ一致回路であり、このデータ一致回路29はデータ比較レジスタ25の内容とデータバス5b上の信号(D15～D0)を随時比較し、両者の内容が一致する場合にDEQ信号を“H”レベルにするものである。ただし、デバッグ制御レジスタ(0)21の内容によって、データ比較レジスタ25の内容をアドレス比較レジスタ(0)23の内容でマスクする機能が有効なモードを選択した場合には、アドレス比較レジスタ(0)23の第15ビット～第0ビットの内の

“1”が設定されたビットに対応するデータ比較レジスタ25のビット内容は、このデータ一致回路29で比較される時にマスクされる。
【0027】30は前述のイベント検出回路であり、このイベント検出回路30は、デバッグ制御レジスタ(0)21およびデバッグ制御レジスタ(1)22から

30 イベント検出の許可/禁止、種類/条件の情報を受け取り、この情報に基づいてアドレス一致比較回路(0)27、アドレス一致比較回路(1)28、およびデータ一致回路29からのAOEQ、AOGT、A1EQ、A1LT、DEQの各信号、およびCPU1から送られてくるROPC、RDA、RDT、RWCPUの各モニタ信号を受け取り、それに基づいてCPU1の動作が指定されたイベントに合致したと判断するとDBC信号を“L”レベルにして、CPU1に対するデバッグ割り込みを発生するものである。

【0028】31はコードアドレスラッチ回路であり、このコードアドレスラッチ回路31は20ビット長のラッチ回路にて形成されており、CPU1から送られてくるアドレス信号のうち、CPU1が現在実行しているプログラムの番地を示すコードアドレスを一時的に蓄えるものである。なお、このコードアドレスラッチ回路31の内容は、トレース機能が有効な時に、後述するトレース制御回路からの指示でGDバス8aに出力され、トレース情報としてICE14に送られる。

【0029】32はデータアドレスラッチ回路であり、このデータアドレスラッチ回路32も20ビット長のラ

(6)

特開平10-214201

9

10

ッチ回路にて形成されており、CPU1から送られてくるアドレス信号のうち、CPU1が現在実行しているデータのリード/ライトの番地を示すデータアドレスを一時的に蓄えるものである。なお、このデータアドレスラッチ回路32の内容も、トレース機能が有効な時に、後述するトレース制御回路からの指示でGDバス8aに出力され、トレース情報としてICE14に送られる。

【0030】33はデータラッチ回路であり、このデータラッチ回路33は16ビット長のラッチ回路にて形成されており、データバス5b上に出力されるCPU1の読み出しデータもしくは書き込みデータを一時的に蓄えるものである。なお、このデータラッチ回路33の内容は、トレース機能が有効な時に、後述するトレース制御回路からの指示でGDバス8aに出力され、トレース情報としてICE14に送られる。

【0031】34はトレース制御回路であり、このトレース制御回路34はデバッグ制御レジスタ(0)21およびデバッグ制御レジスタ(1)22に設定された内容にしたがって、CPU1の動作によって変化するアドレスバス5aの内容(A19~A0)およびデータバス5bの内容(D15~D0)を、一旦コードアドレスラッチ回路31、データアドレスラッチ回路32、あるいはデータラッチ回路33のラッチに蓄え、適宜GDバス8aからICE14にトレース情報として出力する制御を行うものである。

【0032】35はレジスタ制御回路であり、このレジスタ制御回路35は内部にコマンドラッチ回路を備えており、ICE14からGDバス8aを介して、デバッグ制御レジスタ(0)21、デバッグ制御レジスタ(1)22、アドレス比較レジスタ(0)23、アドレス比較レジスタ(1)24、データ比較レジスタ25、あるいはデバッグレジスタ26の内容を読み書きする制御を行うものである。

【0033】36はダウンロード制御回路であり、このダウンロード制御回路36は内部にデータラッチ回路、アドレスラッチ回路(H)、アドレスラッチ回路(L)、コマンドラッチ回路、デコード回路等を備え、ICE14からGDバス8aを介して送られてくるプログラムを、内部バス5に接続されたフラッシュメモリ6に書き込むためのダウンロード機能を制御するものである。

【0034】次にその動作を、内蔵したデバッグ回路7の動作と、そのデバッグ回路7を用いたプログラムのデバッグ処理動作についてそれぞれ説明する。なお、デバッグ回路7の動作については、CPU1とデバッグ回路7との通信、ICE14とデバッグ回路7との通信、ブレーク機能、トレース機能、およびダウンロード機能に分けてそれぞれ説明する。

【0035】まず、デバッグ回路7の動作について、CPU1とデバッグ回路7との通信に関する説明を行う。

マイコン11に内蔵されたデバッグ回路7内のデバッグ制御レジスタ(0)21、デバッグ制御レジスタ(1)22、アドレス比較レジスタ(0)23、アドレス比較レジスタ(1)24、データ比較レジスタ25、およびデバッグレジスタ26は、前述のようにCPU1から読み出し/書き込みが可能に構成されている。これらの各レジスタ21~26にはメモリ空間内の特定アドレスが割り振られており、CPU1が待つ通常の読み出しあるいは書き込み命令によって、これら各レジスタ21~26のアドレスを指定することにより、その読み出し/書き込みが実行される。なお、この実施の形態1における各レジスタ21~26のアドレス配置を図5に示す。図示の例によれば、デバッグ制御レジスタ(0)21には番地“00066。”が割り当てられ、デバッグ制御レジスタ(1)22には番地“00067。”が割り当てられている。

【0036】ここで、CPU1からのデバッグ回路7の内部の各種レジスタ21~26の書き込み/読み出しのタイミングの一例として、デバッグ制御レジスタ(0)21およびデバッグ制御レジスタ(1)22への書き込み/読み出しのタイミングを図6に示す。なお、図6(a)にはデバッグ制御レジスタ(0)21の読み出しタイミングが、図6(b)にはデバッグ制御レジスタ(1)22への書き込みタイミングがそれぞれ示されている。

【0037】図において、φはマイコン11を動作させる基準クロック、A19~A0はアドレスバス5a上の番地であり、DRLE^{*}、DRHE^{*}、DWLE^{*}、およびDWE^{*}は制御バス5c上の制御信号、D7~D0はデータバス5bの下位側のデータ、D15~D8は同じくデータバス5bの上位側のデータである。なお、偶数番地に配置されるレジスタはデータバス5bの下位側(D7~D0)に、奇数番地に配置されるレジスタはデータバス5bの上位側(D15~D8)に接続されている。したがって、番地“00066。”が割り当てられたデバッグ制御レジスタ(0)21はデータバス5bの下位側(D7~D0)に、番地“00067。”が割り当てられたデバッグ制御レジスタ(1)22はデータバス5bの上位側(D15~D8)にそれぞれ接続されている。

【0038】また、前記制御信号中のDRLE^{*}信号とDRHE^{*}信号は読み出しタイミングを示すものであり、DWLE^{*}信号とDWE^{*}信号は書き込みタイミングを示すものである。なお、上記DRLE^{*}信号およびDWLE^{*}信号はデータバス5bの下位側(D7~D0)の読み出し/書き込みのタイミングを示し、DRHE^{*}信号およびDWE^{*}信号はデータバス5bの上位側(D15~D8)の読み出し/書き込みのタイミングを示している。したがって、図6(a)に示すように、読み出しタイミングを示す制御信号DRLE^{*}が“L”レ

(7)

特開平10-214201

11

ベルに変化すると、その時のアドレスバス5 a上の香地
 “00066.” が示すデバッグ制御レジスタ(0) 2
 1より読み出されたデータが、データバス5 bの下位側
 (D7~D0) にてCPU1に送られる。また、図6
 (b) に示すように、書き込みタイミングを示す制御信
 号DWEH' が“L” レベルに変化すると、CPU1か
 らデータバス5 bの上位側 (D15~D8) に送出され
 た書き込みデータが、その時のアドレスバス5 a上の香
 地 “00067.” が示すデバッグ制御レジスタ(1)
 22に書き込まれる。このようにして、マイコン11内
 のCPU1とデバッグ回路7との間で通信が行われる。

【0039】次に、マイコン11の外部に接続されたICE14
 とマイコン11に内蔵されたデバッグ回路7との
 通信について説明する。ICE14とデバッグ回路7
 内の各種レジスタ21~26との間のデータ転送は、デ
 バッグ回路7のGCLK' 端子8 b、GOE' 端子8 c
 およびGDバス8 aとICE14を接続して、ICE14
 側からコマンドを入力して読み出し/書き込みを行う
 方式によって実行される。ここで、GCLK' 端子8 b
 に入力される信号はICE14からの同期クロックであり、
 デバッグ回路7への書き込みにおいてはこの同期ク
 ロックの立ち上がり同期して、GDバス8 aからのデ
 ータがデバッグ回路7内の各種レジスタ21~26に取
 り込まれる。また、デバッグ回路7からの読み出しに
 おいては、GCLK' 端子8 bの同期クロックの立ち下
 がりに同期して、デバッグ回路7内の各種レジスタ21~
 26のデータがGDバス8 aから出力される。また、GOE'
 端子8 cに入力される信号はICE14からのア
 ウトプットイネーブル信号であり、GDバス8 aを介
 してICE14からデバッグ回路7にデータを入力する時
 には“H” レベルにし、デバッグ回路7からICE14
 にデータを読み出す時には“L” レベルにする。

【0040】ここで、ICE14からデバッグ回路7内
 の各種レジスタ21~26を読み出す時のタイミングを
 図7に示す。まずGCLK' 端子8 bのクロックの立ち
 上がりエッジでGDバス8 aからレジスタ読み出しコ
 マンドを入力する。図8はレジスタ読み出し/書き込み
 コマンドのフォーマットを示す説明図で、図示のよう
 に、その上位の3ビット(第7ビット~第5ビット)はコ
 マンドを示すフィールドとなっていて、それが“001”
 のときレジスタ読み出しコマンドであることを示して
 いる。なお、下位の5ビット(第4ビット~第0ビット)
 はアドレスを示すフィールドとなっており、どのレジ
 スタ21~26を読み出すかをこのビットの内容で示して
 いる。

【0041】GDバス8 aから入力されたコマンドはレ
 ジスタ制御回路35に送られ、当該レジスタ制御回路3
 5内のコマンドラッチ回路に格納される。レジスタ制御
 回路35はコマンドラッチ回路に格納されたコマンドの
 第7ビット~第5ビットの“001”をデコードするこ

12

とにより、それがレジスタ読み出しコマンドであるこ
 とがわかる。次にICE14はGOE' 端子8 cの信号を
 “L” レベルにしてGDバス8 aを出力状態にし、続い
 てGCLK' 端子8 bの同期クロックを“L” レベルに
 する。これを受けたレジスタ制御回路35はこの同期ク
 ロックの立ち下がりに同期して、コマンドラッチ回路の
 第4ビット~第0ビットの内容で指定されるレジスタ2
 1~26に対して読み出し信号(#7、#9、#11、
 #13、#15、#17)を送る。これにより指定され
 たレジスタ21~26の内容がGDバス8 aに読み出さ
 れて、外部に接続されたICE14に送られる。

【0042】次に、ICE14からデバッグ回路7内の
 各種レジスタ21~26にデータを書き込む時のタイ
 ミングを図9に示す。なお、図9(a)はデバッグ制御
 レジスタ(0) 21またはデバッグ制御レジスタ(1) 2
 2にデータを書き込む場合のタイミングを示し、図9
 (b)はそれ以外アドレス比較レジスタ(0) 23、
 アドレス比較レジスタ(1) 24、データ比較レジスタ
 25、およびデバッグレジスタ26にデータを書き込む
 場合のタイミングを示している。

【0043】ここで、デバッグ制御レジスタ(0) 21
 またはデバッグ制御レジスタ(1) 22にデータを書き
 込む場合には、GDバス8 aから1回のデータ入力
 で書き込みが行われ、それ以外の各レジスタ23~26に
 関しては、レジスタ書き込みコマンドの入力と書き込みデ
 ータの入力2回の入力動作によって書き込みが実行され
 る。したがって、このレジスタ書き込みコマンドのフォ
 ーマットは前記図8に示したように、デバッグ制御レ
 ジスタ(0) 21へのレジスタ書き込みコマンドはその第
 7ビットを“1”として下位の6ビットを書き込みデ
 ータのためのフィールドとし、デバッグ制御レジスタ
 (1) 22へのレジスタ書き込みコマンドはその第7ビ
 ットおよび第6ビットを“01”として下位の5ビット
 を書き込みデータのためのフィールドとしている。ま
 た、それ以外の各レジスタ23~26へのレジスタ書き
 込みコマンドはその第7ビット~第5ビットを“00
 0”として下位の5ビットを読み出すレジスタ23~2
 6のアドレスを示すフィールドとしている。

【0044】レジスタ制御回路35は図9(a)に示す
 ように、GCLK' 端子8 bの同期クロックの立ち上
 がりエッジでGDバス8 aからコマンドが入力されると、
 それをコマンドラッチ回路に格納してその内容のデコ
 ードを行う。その結果、第7ビットが“1”であれば、
 それがデバッグ制御レジスタ(0) 21へのレジスタ書き
 込みコマンドであると判断して、デバッグ制御レジスタ
 (0) 21に書き込み信号(#14)を送る。これによ
 り、GDバス8 a上の下位7ビット(GD6~GD0)
 の内容がデバッグ制御レジスタ(0) 21に書き込ま
 れる。同様に、第7ビットおよび第6ビットが“0
 1”であれば、そのコマンドがデバッグ制御レジスタ

(8)

特開平10-214201

13

(1) 22へのレジスタ書き込みコマンドであると判断して、デバッグ制御レジスタ(1) 22に書き込み信号(#16)を送る。これにより、GDバス8aの下位6ビット(GD5~GD0)の内容がデバッグ制御レジスタ(1) 22に書き込まれる。このようにして、デバッグ制御レジスタ(0) 21およびデバッグ制御レジスタ(1) 22への書き込みは、GDバス8aからの1回の入力で実行される。

【0045】一方、レジスタ制御回路35内のコマンドラッチ回路の第7ビット~第5ビットの内容が“000”であった場合には、そのコマンドはデバッグ制御レジスタ(0) 21およびデバッグ制御レジスタ(1) 22以外のレジスタ23~26へのレジスタ書き込みコマンドであり、その第4ビット~第0ビットがどのレジスタ23~26に書き込むかのアドレス情報であると判断し、次の書き込みデータの待機を待つ。ICE14は図9(b)に示すように、GOE' 端子8cの信号を“L”レベルにして、GCLK' 端子8bの同期クロックを“L”レベルにし、この同期クロックの立ち下がり同期して書き込みデータをGDバス8aに送出する。書き込みデータがGDバス8aに入力されると、レジスタ制御回路35はこの同期クロックの立ち下がり同期して、コマンドラッチ回路の第4ビット~第0ビットの内容によって指定されるレジスタ23~26に対して、書き込み信号(#8、#10、#12、あるいは#18)を送る。このようにして、マイコン11の外部に接続されたICE14とマイコン11に内蔵されたデバッグ回路7との間で通信が行われる。

【0046】次にブレーク機能の動作について説明する。ここで、ブレーク機能とはマイコン11のCPU1がユーザプログラムを実行する状態を、マイコン11に内蔵されたデバッグ回路7が監視し、CPU1の動作状態がデバッグ制御レジスタ(0) 21およびデバッグ制御レジスタ(1) 22に設定した状態と合致した時に、デバッグ割り込みを発生させてユーザプログラムを中断させる機能である。

【0047】図4(a)に示すように、デバッグ制御レジスタ(0) 21の第5ビットはブレークモード許可ビットであり、ブレーク機能はそれを“1”にセットすることによって有効になり、“0”にセットすることによって無効となる。また、第2ビット~第0ビットはブレーク検出条件選択ビットで、このビットの内容によってブレークを検出する条件の選択を行う。なお、ブレーク検出条件はアドレス・データ一致検出、アドレス一致検出(0)、アドレス一致検出(1)、アドレス一致検出(2)、アドレス範囲内検出、アドレス範囲外検出のうちからその1つを選択することができる。以下、その各々について順次説明する。

【0048】アドレス・データ一致検出はデバッグ制御レジスタ(0) 21の第2ビット~第0ビットの内容が

14

“000”の時に選択される。この場合、CPU1がデータの読み出しあるいは書き込みを行う際に、その読み出し/書き込みアドレスとアドレス比較レジスタ(1) 24の内容が一致し、かつ読み出しあるいは書き込みデータがデータ比較レジスタ25の内容と一致した時、デバッグ割り込みを発生させる。先に述べたように、このデータ比較レジスタ25の内容はアドレス比較レジスタ(0) 23の内容にてマスクすることができる。すなわち、アドレス比較レジスタ(0) 23の下位16ビットがデータ比較レジスタ25の各ビットに対応し、アドレス比較レジスタ(0) 23のビットが“1”の時、データ比較レジスタ25の対応するビットがマスクされる。

【0049】アドレス一致検出(0)はデバッグ制御レジスタ(0) 21の第2ビット~第0ビットの内容が“001”の時に選択される。この場合、CPU1からの出力アドレスがアドレス比較レジスタ(0) 23の内容と一致した時にデバッグ割り込みを発生させる。

【0050】アドレス一致検出(1)はデバッグ制御レジスタ(0) 21の第2ビット~第0ビットの内容が“010”の時に選択される。この場合、CPU1からの出力アドレスがアドレス比較レジスタ(1) 24の内容と一致した時にデバッグ割り込みを発生させる。

【0051】アドレス一致検出(2)はデバッグ制御レジスタ(0) 21の第2ビット~第0ビットの内容が“011”の時に選択される。この場合、CPU1からの出力アドレスがアドレス比較レジスタ(0) 23またはアドレス比較レジスタ(1) 24の内容と一致した時にデバッグ割り込みを発生させる。

【0052】アドレス範囲内検出はデバッグ制御レジスタ(0) 21の第2ビット~第0ビットの内容が“100”の時に選択される。この場合、CPU1からの出力アドレスがアドレス比較レジスタ(0) 23の内容以上であり、かつアドレス比較レジスタ(1) 24の内容以下である時にデバッグ割り込みを発生させる。

【0053】アドレス範囲外検出はデバッグ制御レジスタ(0) 21の第2ビット~第0ビットの内容が“101”の時に選択される。この場合、CPU1からの出力アドレスがアドレス比較レジスタ(0) 23の内容よりも小さいか、アドレス比較レジスタ(1) 24の内容よりも大きい時にデバッグ割り込みを発生させる。

【0054】また、デバッグ制御レジスタ(0) 21の第4ビット~第3ビットはイベント検出サイクル選択ビットで、このビットの内容によってブレークを検出するイベントの種類を選択を行う。なお、イベントの種類はオペコードアドレス、データ読み出し、データ書き込み、データ読み出し・書き込みのうちからその1つを選択することができる。以下、その各々について順次説明する。

【0055】オペコードアドレスはデバッグ制御レジスタ(0) 21の第4ビット~第3ビットの内容が“0

15

0”の時に選択される。この場合には、CPU1が命令の第1バイト、すなわちオペコードを読み出す時のアドレスが、デバッグ制御レジスタ(0)21の第2ビット～第0ビットにおけるブレーク検出条件選択ビットで選択された条件の検出対象となる。

【0056】データ読み出しはデバッグ制御レジスタ(0)21の第4ビット～第3ビットの内容が“01”の時に選択される。この場合、CPU1がデータを読み出す時のアドレスまたはデータが、デバッグ制御レジスタ(0)21の第2ビット～第0ビットにおけるブレーク検出条件選択ビットで選択された条件の検出対象となる。

【0057】データ書き込みはデバッグ制御レジスタ(0)21の第4ビット～第3ビットの内容が“10”の時に選択される。この場合、CPU1がデータを書き込む時のアドレスまたはデータが、デバッグ制御レジスタ(0)21の第2ビット～第0ビットにおけるブレーク検出条件選択ビットで選択された条件の検出対象となる。

【0058】データ読み出し・書き込みはデバッグ制御レジスタ(0)21の第4ビット～第3ビットの内容が“11”の時に選択される。この場合、CPU1がデータ転送命令などにおけるデータの読み出し/書き込みを連続して行う場合のアドレスまたはデータが、デバッグ制御レジスタ(0)21の第2ビット～第0ビットにおけるブレーク検出条件選択ビットで選択された条件の検出対象となる。

【0059】また、先に述べたように、アドレス比較レジスタ(0)23、アドレス比較レジスタ(1)24、およびデータ比較レジスタ25の内容は、アドレス一致比較回路(0)27、アドレス一致比較回路(1)28、およびデータ一致回路29によってCPU1から入出力されるアドレスバス5a上のアドレス信号(A19～A0)およびバス5b上のデータ信号(D15～D0)とリアルタイムに、その一致あるいは大小関係が判定されており、その結果が信号A0EQ、A0GT、A1EQ、A1LT、DEQとしてイベント検出回路30に送られる。イベント検出回路30ではこれらの各信号と、CPU1から送られてくるその動作状態を表すモニタ信号ROP、RDA、RDT、RWCPUとから、デバッグ制御レジスタ(0)21で設定されたイベントの種類/検出条件が満足されるか否かを逐次監視し、満足した時にはDBC'信号を“L”レベルにしてCPU1に対してデバッグ割り込みを要求する。

【0060】なお、上記ROP信号はCPU1がオペコードを読み出す時に出力される信号であり、RDA信号はCPU1がデータのアクセスアドレスを出力する時に出力される信号、RDT信号はCPU1が読み出したデータをとり込む時に出力される信号、RWCPU信号はCPU1のデータアクセスが読み出し(“H”レベ

(9)

特開平10-214201

16

ル)か、書き込み(“L”レベル)かを示す信号である。

【0061】以上のような、CPU1の動作状態を監視してあらかじめ設定された条件に一致した時にデバッグ割り込みを発生させる以外に、このCPU1の動作とは無関係に、マイコン11の外部に接続されたICE14より任意のタイミングでデバッグ割り込みの発生を要求する機能も持っている。すなわち、ICE14からデバッグ制御レジスタ(1)22への書き込みコマンドを用いて、その第4ビットを“1”にセットすると、イベント検出回路30ではCPU1に対しデバッグ割り込みを要求する。

【0062】次にトレース機能の動作について説明する。図4(a)に示すように、デバッグ制御レジスタ(0)21の第6ビットはトレースモード許可ビットであり、このビットを“1”にセットすることによってトレース機能が有効になる。トレースモード許可ビットがセットされ、かつGOE'端子8cに入力される信号が“L”レベルになると、トレース制御回路34では後述するトレース出力すべき条件を検出することに、GCLK'端子8bからクロックを出力し、これに同期させてGDバス8aから8ビット単位でトレースデータを出力する。なお、このトレースデータにはコードトレースとデータトレースの2種類がある。

【0063】コードトレースはCPU1の動作に応じて、以下の内容のアドレス情報をトレースデータとして出力するものである。すなわち、

- (1) リセット解除後のプログラム実行開始番地
- (2) 割り込み要求受け付け時のオペコードアドレスおよび割り込み処理のプログラム実行開始番地
- (3) 分岐命令のオペコードアドレスおよび分岐先番地

の3種類である。

【0064】このトレースデータ(アドレス情報)は3バイト構成となっており、図10に示すように、GCLK'端子8bのクロックに同期して上位アドレス、中位アドレス、下位アドレスの順に出力する。なお、1バイト目に出力される上位アドレスの上位4ビット(第7ビット～第4ビット)はトレース属性であり、図11に示すように、以下に続くアドレス情報が上記(1)、

(2)、(3)のいずれであるかを示している。また、その下位4ビット(第3ビット～第0ビット)はアドレスA19～A16であり、2バイト目に出力される中位アドレスはアドレスA15～A8、3バイト目に出力される下位アドレスはアドレスA7～A0である。

【0065】このデータトレースは図12あるいは図13に示すように、GCLK'端子8bのクロックに同期して、CPU1がデータを読み出したり書き込んだりする時のアドレスとデータを、5バイトまたは8バイトのデータで出力するものである。図12はデータの読み出

50

(10)

特開平10-214201

17

し/書き込みの場合のトレース出力例を示しており、この場合は通常5バイトのトレースデータが出力される。すなわち、その最初の3バイトで読み出しアドレスあるいは書き込みアドレスの上位アドレス、中位アドレス、および下位アドレスが、次の2バイトで読み出しデータあるいは書き込みデータの上位データと下位データが順番に出力される。なお、1バイト目の上位4ビットはコードトレースの場合と同様にトレース属性であり、図11に示すようにデータの読み出し/書き込みの種類を表している。

【0066】また、図13にはデータの読み出しと書き込みを連続して行うデータ転送命令の場合のトレース出力例を示しており、この場合には8バイトのトレースデータが出力され、その最初の3バイトで転送元アドレスの上位アドレス、中位アドレスおよび下位アドレスが、それに続く3バイトで転送先アドレスの上位アドレス、中位アドレスおよび下位アドレスが、最後の2バイトで転送データの上位データ、下位データが順番に出力される。なお、この場合も、転送元アドレスおよび転送先アドレスの上位アドレスにおける上位4ビットは、コード

トレースの場合と同様にトレース属性であり、図11に示すようにデータの読み出し/書き込みの種類を表している。

【0067】ここで、トレース出力を停止させてICE14側からGDバス8aを介してデバッグ回路7と通信したい場合には、ICE14はまずGOE'端子8cの信号を"H"レベルにする。このGOE'端子8cの信号が"H"レベルになるとトレース制御回路34は、その時点で実行中のトレース出力を完了した後、GCLK'端子8bのクロックの1回の立ち下がりに同期させて、GDバス8aに"4X₀"という認識コードを送る。その後、GCLK'端子8bとGDバス8aを出力から入力に切り替える。ICE14はこの"4X₀"という認識コードを受け取ることで、GCLK'端子8bとGDバス8aが入力状態になったことを知り、以後、前述のICE14とデバッグ回路7との通信動作において説明した方法によって、デバッグ回路7の各レジスタ21~26に対して読み出し/書き込みを実行する。

【0068】また、前述のブレーク機能を使用する場合には、通常、トレース機能も有効にする。すなわち、ユーザプログラムを実行中にイベント検出回路30によってブレーク条件を検出し、デバッグ割り込み要求が発生すると、トレース制御回路34は現在実行中のトレース出力完了後、GCLK'端子8bのクロックの1回の立ち下がりに同期させて、GDバス8aに"8X₀"という認識コードを出力し、ICE14に対してブレーク検出によるデバッグ割り込みの発生によってユーザプログラムの実行が中断されたことを知らせる。

【0069】次にダウンロード機能の動作について説明

18

する。ダウンロードは、マイコン11に内蔵されたフラッシュメモリ6に対して、リード/プログラム/イレースなどの操作を、GDバス8aを介して行う機能である。デバッグ回路7は以下の手順に従って、このダウンロードを実行するためのダウンロードモードに移行する。すなわち、まずRESET'端子8eを"L"レベルにしてマイコン11をリセット状態にする。このリセット状態においては、マイコン11のCPU1は内部のアドレスバス5a、データバス5bから切り離され、これに代わってダウンロード制御回路36が接続される。次に、デバッグ制御レジスタ(1)22の第0ビットを"1"に設定し、その後Vpp端子8dを0Vから12Vにする。これによってデバッグ回路7はダウンロードモードに移行する。

【0070】なお、このダウンロードモードにおいては、ICE14からデバッグ回路7のダウンロード制御回路36を介して、マイコン11内のフラッシュメモリ6に対するリード、プログラム、プログラムベリファイ、およびオートイレースの4つのコマンドを発行することができる。

【0071】フラッシュメモリ6自体はすでに実用化されている公知のものであり、その入出力信号を図14に示す。このフラッシュメモリ6の読み出し/書き込み/消去動作は、図14に示すアドレス(A15~A0)入力、データ(D7~D0)入出力、CE'、OE'、WE'の各制御入力、Vpp入力、およびビジー(BUSY)出力によって操作される。したがって、デバッグ回路7内のダウンロード制御回路36の役割は、ICE14よりGDバス8aを経由して8ビット単位で送られてくるデータを、マイコン11内部のアドレスバス5aおよびデータバス5bを介してフラッシュメモリ6に送るとともに、CE'、OE'およびWE'の各信号を必要なタイミングで生成して、同じくフラッシュメモリ6へ送ることである。以下、リード、プログラム、プログラムベリファイ、オートイレースの各コマンド実行時における、ICE14からのデータ入出力タイミング、およびダウンロード制御回路36とフラッシュメモリ6間の信号の動作タイミングについて説明する。

【0072】まずリードコマンド実行時の動作について説明する。ここで、図15はリードコマンド実行時のダウンロード制御回路36の動作タイミングを示すタイミングチャートである。まず最初に、ICE14側からGCLK'端子8bのクロックに同期して、GDバス8aよりコマンドコード(リードコマンドの場合は"00₀₀")を入力する。ダウンロードモードの場合、このコマンドコードはダウンロード制御回路36内のデータラッチ回路にラッチされるとともに、データバス5bの下位側(D7~D0)に出力される。またダウンロード制御回路36ではGCLK'端子8bのクロックに同期したWE'信号を生成してフラッシュメモリ6に送る。フ

(11)

特開平10-214201

19

20

ラッシュメモリ6はデータバス5bの下位側(D7~D0)上に出力されたこのコマンドコードを、このWE'信号に同期してラッチする。

【0073】また同時に、ダウンロード制御回路36ではデータラッチ回路の内容がコマンドラッチ回路よりデコード回路に送られてデコードされ、当該コマンドがリードコマンドであることが解釈される。このデコード結果に基づいて、ICE14からの2回目の入力(リードアドレスの下位側)と判断され、それがアドレスラッチ回路(L)にラッチされるとともに、アドレスバス5aの下位側(A7~A0)に出力されてフラッシュメモリ6に入力される。さらにICE14からの3回目の入力はリードアドレスの上位側と判断されてアドレスラッチ回路(H)にラッチされるとともに、アドレスバス5aの上位側(A15~A8)に出力され、同様にフラッシュメモリ6に入力される。

【0074】続いて、ICE14はGOE'端子8cの信号を"L"レベルにしてGDバス8aを出力状態にした後、GCLK'端子8bからのクロックを入力する。ダウンロード制御回路36ではこのクロックに同期してOE'信号を生成し、それをフラッシュメモリ6に送る。フラッシュメモリ6はこのOE'信号に同期して入力されているアドレスで指定された番地のデータをデータバス5bの下位側(D7~D0)に出力する。このデータはさらにGDバス8aに出力されてICE14に送られる。以上の操作によってICE14はマイコン11に内蔵されたフラッシュメモリ6の指定番地のデータを読み出せた、すなわちリードコマンドの実行を完了したことになる。

【0075】次にプログラムコマンド実行時の動作について説明する。ここで、図16はプログラムコマンド実行時のダウンロード制御回路36の動作タイミングを示すタイミングチャートである。なお、動作の基本的な考え方はリードコマンド実行時と同様である。すなわち、1回目にはコマンドコードを、2回目にはプログラムアドレスの下位側を、3回目にはプログラムアドレスの上位側を順次入力してそれらをフラッシュメモリ6に送る。なお、このプログラムコマンド実行時にはさらに、4回目の入力でプログラムデータを外部のICE14側よりデバッグ回路7に送る。このプログラムデータはダウンロード制御回路36のデータラッチ回路にラッチされるとともに、データバス5bの下位側(D7~D0)に出力される。ダウンロード制御回路36ではGCLK'端子8bの4回目のクロックに同期してWE'信号を生成し、それをフラッシュメモリ6に送る。

【0076】フラッシュメモリ6はこのWE'信号の立ち上がりエッジから、アドレス入力されている番地にデータバス5bの下位側(D7~D0)上のデータをプログラムする動作を開始する。プログラム動作が始まるとフラッシュメモリ6から出力されるBUSY信号が

"H"レベルになる。このBUSY信号はプログラムコマンド実行期間中は"H"レベルを維持し、終了とともに"L"レベルに戻る。このBUSY信号はデバッグ制御レジスタ(1)22の第1ビットに入力されており、ICE14はプログラムコマンド発行後、このビットの読み出しを繰り返すことによってプログラムコマンドの終了を知ることができる。

【0077】次にプログラムベリファイ実行時の動作について説明する。ここで、図17はプログラムベリファイ実行時のダウンロード制御回路36の動作タイミングを示すタイミングチャートである。なお、プログラムベリファイコマンドはプログラムコマンドと必ず対で実行される。プログラムアドレスはプログラムコマンド実行時に、フラッシュメモリ6の内部回路でラッチされているため、プログラムベリファイコマンド実行時にはベリファイアドレスをあらためて入力する必要がない。

【0078】次にオートイレース実行時の動作について説明する。ここで、図18はオートイレースコマンド実行時のダウンロード制御回路36の動作タイミングを示すタイミングチャートである。まずICE14より、コマンドコード"30."を続けて2回入力する。このコマンドコードはいずれもダウンロード制御回路36のデータラッチ回路よりデータバス5bの下位側(D7~D0)を介してフラッシュメモリ6に送られる。またダウンロード制御回路36はGCLK'端子8bのクロックの2回目の入力に同期してWE'信号を生成し、それをフラッシュメモリ6に送る。フラッシュメモリ6では2回目のWE'信号の立ち上がりエッジに同期して消去動作がスタートする。なお、この消去動作中はBUSY信号が"H"レベルになるため、プログラムコマンド実行時と同様に、ICE14はこれを監視することによって消去動作の完了を知ることができる。

【0079】次に、これまでに説明したデバッグ回路7を用いて、ユーザが実際にプログラムをデバッグする際のデバッグ処理動作について説明する。ここで、プログラムのデバッグを実行する際のシステム構成は図2に示す通りである。ユーザはまず、パソコンあるいはEWSなどによるホストコンピュータ15上で、C言語、アセンブラ言語などを用いてプログラムを作成する。プログラムがとりあえずでき上がると、そのプログラムをコンパイルして16進コードのファイルに変換し、それをICE14に転送する。このICE14はユーザ基板16上に直接搭載されたマイコン11にコネクタ17を介してケーブル接続されており、転送されてきたコンパイル済のユーザプログラムを、そのマイコン11に内蔵されたフラッシュメモリ6に、デバッグ回路7のダウンロード機能を利用して書き込む。

【0080】ここで、デバッグ割り込みが発生した時に実行させるユーザプログラムとは別の、デバッグ専用のプログラム(以後デバッグモニタといい、その役割につ

(12)

特開平10-214201

21

いては後述する)を用意しておき、ユーザプログラムと一緒にあらかじめ定めたフラッシュメモリ6の特定領域にダウンロードしておく。すなわちフラッシュメモリ6に書き込まれるのは、ユーザプログラムとデバッグ割り込みが発生した時に実行するデバッグモニタである。

【0081】ICE14はダウンロードが完了すると、Vpp端子8dをVssレベル(0V)に戻してダウンロードモードを解除する。ただし、RESET⁺端子8eの信号は“L”レベルのままとしてマイコン11はリセット状態にしておく。この状態でICE14はGDバス8aを介して、マイコン11に内蔵されたデバッグ回路7内のデバッグ制御レジスタ(0)21、デバッグ制御レジスタ(1)22、アドレス比較レジスタ(0)23、アドレス比較レジスタ(1)24、データ比較レジスタ25等に必要な条件を設定する。

【0082】例えば、リセット解除後のプログラムの実行動作を少しずつ区切って正しく動作しているかどうかを確かめたい場合には、まずトレース機能を有効にし、さらにリセット解除後、プログラムの実行がある程度進んだところにブレークポイント(オペコードアドレスによるブレーク)を設定する。このような条件設定をした後、ICE14からRESET⁺端子8eを“H”レベルに戻すと、マイコン11はリセット解除されてユーザプログラムの実行を開始する。そして最初に設定したブレークポイントのところまでユーザプログラムの実行が進むと、ブレーク機能の働きによってデバッグ割り込みが発生し、ユーザプログラムの実行は中断されてCPU11はデバッグモニタの実行に移る。

【0083】ユーザプログラムが中断されるまでのCPU11の動作のトレース情報は、トレース機能が有効にされているため、GDバス8aからICE14へ順次送られてきており、ICE14内のトレースメモリに記憶されている。したがって、ホストコンピュータ15側からこれを読み出して表示させれば、ユーザが自分の作成したプログラムが、リセット解除後ブレークポイントを設定したところまで、正しく動作しているかどうかをチェックすることができる。

【0084】例えば、トレース結果から判断すると、ユーザプログラムを中断したブレークポイントの直前で、マイコン11の動作、すなわちCPU1の動作がユーザの意図したものとは異なった動作をしている場合を想定してみる。ユーザはこのような状態で、ホストコンピュータ15からICE14を介して種々のコマンドをマイコン11に発行してその原因を探る。これにより、作成したユーザプログラムのデバッグ作業を行うことが可能となる。その際に、先にふれたデバッグモニタと呼ばれるプログラムが使用される。デバッグ割り込みが発生すると、CPU11はユーザプログラムの実行を中断してデバッグモニタの先頭から実行を開始する。デバッグモニタの一例を図19のフローチャートに示す。

22

【0085】デバッグ割り込みが発生してデバッグモニタに実行に移ると、CPU11はまず、デバッグ回路7のデバッグ制御レジスタ(1)22の第2ビットのサービスリクエストフラグをチェックして(ステップST1)、その内容が“0”の場合にはこのフラグが“1”になるまでチェックのループを繰り返す。

【0086】一方、デバッグ割り込みが発生すると、デバッグ回路7のトレース制御回路34は、その時点で実行中のトレース出力完了後にブレーク認識コード“8X”を出力する。これによりICE14は、イベント検出によってブレークが発生し、CPU11がユーザプログラムの実行からデバッグモニタの実行に移ったことを知る。続いてICE14はGOE⁺端子8cの信号を“H”レベルに戻す。デバッグ回路7はGOE⁺→“H”の認識コード“4X”を出力した後、GCLK⁺端子8bおよびGDバス8aを入力に切り替える。ICE14はこの認識コード“4X”を受け取ることににより、自分がGDバス8a、GCLK⁺端子8b、およびGOE⁺端子8cを制御してデバッグ回路7内の各種レジスタ21～26にアクセスできる状態になったことを知る。

【0087】この状態で、ICE14はユーザがユーザプログラムをデバッグするための種々の操作(コマンド)を提供することができる。一例として、ユーザプログラムがブレークによって中断した状態で、マイコン11内部のある番地のデータ(例えばRAM3のある番地に格納されているデータ)の内容を見たい場合について考える。この場合、ICE14はまず、メモリ読み出しのためのコマンドを、デバッグ回路7のデバッグレジスタ(0)26に書き込む。なお、このコマンドを“00”～“FF”のどれにするかは、ICE14とデバッグモニタの間で取り決めておけばよい。続いてデバッグレジスタ(1)、(2)および(3)26にそれぞれ読み出したいデータの低位アドレス、中位アドレスおよび上位アドレスを書き込む。次にICE14は、デバッグ制御レジスタ(1)22の第2ビット(サービスリクエストフラグ)を“1”にセットし、その後デバッグ制御レジスタ(1)22の第3ビット(サービスエンドフラグ)の内容のチェックを繰り返す。これが図19における、サービスリクエストフラグが“1”になるまで待つループとなる。

【0088】サービスリクエストフラグが“1”になるのを待っていたCPU11は、上記したICE14の操作によってその内容が“1”にセットされると、当該チェックループから抜け出してこのリクエストフラグをクリアし(ステップST2)、その後デバッグレジスタ(0)26の内容を読み出す(ステップST3)。次にその内容をチェック(読み出したコマンドをデコード)することによってICE14が要求するサービスの内容を解釈し、それを実行するルーチンにジャンプする(ス

(13)

特開平10-214201

23

24

テップST4)。

【0089】この例のメモリ読み出しコマンドの場合には、まず、デバッグレジスタ(1)、(2)および(3)26に格納されているアドレス情報を取り出し(ステップST5)、そのアドレスによって指定された番地の内容を読み出す(ステップST6)。続いてその読み出したデータを下位側と上位側に分けて、デバッグレジスタ(4)および(5)26にそれぞれ格納する(ステップST7)。指定されたコマンドの実行を完了すると、サービスエンドフラグを“1”にセットし(ス

テップST8)、その後ステップST1に戻って、サービスリクエストフラグのチェックループに入る。
【0090】このサービスエンドフラグが“1”にセットされるのを待っていたICE14は、当該サービスエンドフラグが“1”にセットされることにより、要求したサービス(メモリ読み出しコマンド)が完了したことを知り、このフラグをクリアした後、デバッグ回路7内のデバッグレジスタ(4)および(5)26からデータを読み出す。以上のような一連の動作によって、「メモリ読み出し」という一つの操作が完了することになる。CPU1内部のレジスタの内容を読み出したり、あるいはレジスタ/メモリに指定されたデータを書き込むコマンドなどもこれと同様の手順で実現することができる。

【0091】また、ICE14はその他に、“GO”コマンドおよび“STOP”コマンドも提供する。デバッグモニタがサービスリクエストの待ちループの中にある時に“GO”コマンドを指定すると、デバッグ割り込みからの復帰コマンドを実行して(図19のステップST9)デバッグモニタの実行は終了し、CPU1はユーザプログラムの実行を中断した状態から再開する。一方、“STOP”コマンドはCPU1がユーザプログラムを実行している際に、イベント検出によるブレーク機能ではなく、ICE14側から強制的に中断させてデバッグモニタを起動させるコマンドである。このコマンドはICE14がGDバス8a経由でデバッグ回路7のデバッグ制御レジスタ(1)22の第4ビット(デバッグ割り込み要求ビット)に“1”を書き込むことによって実現される。

【0092】以上のように、デバッグレジスタ26を仲介して、マイコン11の外部に接続されたICE14とデバッグモニタとがハンドシェイクすることにより、ユーザがプログラムをデバッグするための種々の機能が提供される。

【0093】このように、この実施の形態1によれば、マイコンに内蔵したフラッシュメモリ6をエミュレーションメモリとして使用するため、ROM2の場合と同等の速度でアクセスすることができ、外部にエミュレーションメモリを設ける従来のマイコンのようなアクセス速度の問題が解消され、また、マイコンに特別なデバッグ専用のモードを設ける必要がなくなるばかりか、マイコ

ン11をユーザ基板16に直付けし、デバッグに關係する信号だけをコネクタ17を介してICE14に接続させる形態になって、マイコン11の入出力信号のタイミングがケーブルによる影響を受けることがなくなるため、極めて最終製品の真装状態に近い形でプログラムの開発/デバッグを行うことができ、また、イベント検出によるブレーク機能をマイコン11のチップ内部に持っているため、あらかじめ設定されたイベントを検出すると、直ちにデバッグ割り込みを発生しユーザプログラムの実行を中断することができるのでブレーク発生の遅れがなくなるなどの効果がある。

【0094】さらに、この実施の形態1によれば、GDバス8a上の信号およびGDLK'端子8bの信号をGOE'端子8cの信号レベルによって双方向に使用することができるようになり、外部のICE14からのデバッグ割り込み要求、マイコン11内部のデバッグ回路7からのイベント検出によるブレーク発生の認識信号出力、GOE'→“H”の認識コード出力等もすべてGDバス8a経由で行えるため、ICE14とデバッグ回路7との間の信号線を最低本数に抑えることができる効果もある。

【0095】実施の形態2。先に述べたように、マイコン11に内蔵されているROM2は、その内容が半導体の製造工程中に書き込まれてしまい、それ以降は書き換えができない。したがって、製造完了後にROM2に書き込んだプログラムのバグが見つかったと、製品そのものが使えなくなるというリスクを抱えている。上記実施の形態1で説明したデバッグ回路7のブレーク機能は、このようなバグのあるプログラム部分の回避に応用することも可能である。図20はそのようなこの発明の実施の形態2によるマイコンに内蔵されたデバッグ回路のブレーク機能を応用して、バグのあるプログラム部分を回避するための「ROMコレクション」の手順を示すフローチャートである。

【0096】まず、このROMコレクションをするか否かのフラグをマイコン11の外部に設けておく。例えば、外部メモリの特定番地の特定ビットが“0”であればROMコレクション不要、“1”であればROMコレクション必要、あるいはマイコン11のある端子が“L”レベルであればROMコレクション不要、“H”レベルであればROMコレクション必要等である。

【0097】図20に示すように、リセットスタート後、上記フラグによってROMコレクションの有効/無効をまずチェックする(ステップST12)。その結果、無効と判定されてROMコレクションの必要がない場合には、そのままマイコン11内のROM2に書かれたプログラムを実行する。一方、有効と判定されてROMコレクションを実行する場合には、ブレーク条件の設定を行う(ステップST13)。図3に示したデバッグ回路7においては、アドレス比較レジスタ(0)23と

(14)

特開平10-214201

25

26

アドレス比較レジスタ(1)24の2本を持っているため、最大2箇所のバグ回過が可能である。したがって、ここでは2箇所のバグを回過する例について説明する。まずバグのあるプログラム部分のそれぞれの先頭番地を外部から読み込み、アドレス比較レジスタ(0)23およびアドレス比較レジスタ(1)24に書き込む。次にデバグ制御レジスタ(0)21において、その第4ビット～第3ビットのイベント検出サイクル選択ビットではオペコードアドレスを選択し、第2ビット第0ビットのブレーク検出条件選択ビットではアドレス一致検出

(2)を選択する。

【0098】プログラムの実行が進み、バグのあるプログラム部分の先頭にくると、ブレーク機能によりデバグ割り込みが発生し、デバグ割り込み処理プログラムに移行する。デバグ割り込み処理プログラムにおいては、まずアドレス比較レジスタ(0)23またはアドレス比較レジスタ(1)24のどちらでデバグ割り込みが発生したかをチェックする(ステップST14)。そのために、デバグ制御レジスタの空きビット、例えばデバグ制御レジスタ(1)21の第5ビットをその判定ビットに割り当て、アドレス比較レジスタ(0)23でブレークが発生した場合にはこの判定ビットをクリアし、比較レジスタ(1)24で発生した場合にはそれをセットするハードウェアを追加する。

【0099】また、アドレス比較レジスタ(0)23の一致検出で割り込みが発生した場合の正しいプログラムが格納されている外部メモリの番地、およびアドレス比較レジスタ(1)24の一致検出で割り込みが発生した場合の正しいプログラムが格納されている外部メモリの番地をあらかじめ決めておき、ステップST14によるアドレス比較レジスタ(0)23またはアドレス比較レジスタ(1)24のどちらでデバグ割り込みが発生したかのチェック結果に基づいて、対応する正しいプログラムが格納されている外部メモリの先頭番地の内容を読み出す(ステップST15)。なお、このデバグ割り込み処理プログラムに移る時に、スタック領域に処理完了後の戻り先番地が格納されているが、その戻り先番地はバグのあるプログラムを指しているため、これをステップST15で読み出した、正しいプログラムの先頭番地に変更する(ステップST16)。その後、デバグ割り込みから復帰する命令が実行されると(ステップST17)、外部メモリにおかれているバグの修正された正しいプログラムの実行に移る。

【0100】実施の形態3、上記実施の形態2においては、マイコンに内蔵されたデバグ回路のブレーク機能を「ROMコレクション」に応用した場合について説明したが、「暴走検出」に応用することも可能である。以下、そのようなこの発明の実施の形態3について説明する。

【0101】マイコン11が何らかの原因によって暴走

状態に至った場合、システムに致命的な打撃を与えないために、従来より監視タイマの機能がよく用いられる。この監視タイマはある一定の周期を計測するタイマで、その周期内に一度もCPU1からアクセスされないと、当該監視タイマ専用の割り込みを発生する機能を持っている。一方、CPU1からアクセスされると監視タイマはその時点で一旦リセットされ、一定周期のカウントをそこから再開する。ユーザは上記一定周期内に監視タイマをアクセスするようにプログラムを作っておく。これにより、プログラムが正常に動作している間は周期的に監視タイマにアクセスするため、監視タイマ専用の割り込みは発生しない。マイコン11が暴走状態に至ると監視タイマのアクセスがなくなるため、監視タイマ割り込みが発生する。割り込みが発生するとそれまでマイコン11が暴走状態にあっても一旦は必ず監視タイマ割り込みプログラムに分岐するため、そこでマイコン11自身をリセットする処置などをして暴走状態から復帰する。

【0102】この実施の形態3は、マイコン11に内蔵されたデバグ回路7のブレーク機能を用いることによって、これと同等、あるいはそれ以上に正確な暴走検出を可能にしたものである。すなわち、デバグ制御レジスタ(0)21の第4ビット～第3ビットによるイベント検出サイクル選択ビットをオペコードアドレスに、第3ビット～第0ビットによブレーク検出条件選択ビットをアドレス範囲外検出にそれぞれ設定する。このように設定しておけば、プログラムがおかれるアドレス範囲はあらかじめわかっているため、暴走状態になってCPU1の実行アドレスがその設定された範囲外に及ぶと直ちにデバグ割り込みが発生するので、そのような場合には割り込み処理プログラムで暴走からの復帰処理を実行すればよい。監視タイマを用いる場合には、暴走状態に到ってもある一定周期を経ないとそれを検出することができないが、この実施の形態3によれば、早期に暴走状態を検出し、それから復帰することが可能となる。

【0103】

【発明の効果】以上のように、請求項1記載の発明によれば、外部のICEとの接続のための専用入出力端子を有したデバグ回路と、開発段階のプログラムが格納されるフラッシュメモリを内蔵して、そのデバグ回路に、CPUとの通信機能、ICEとの通信機能、CPUの動作状態のトレース機能、デバグ専用の割り込みを発生させるブレーク機能、ICEからのプログラムコードをフラッシュメモリに書き込み、フラッシュメモリの内容をICEに送る機能などの諸機能を持たせるように構成したので、内蔵するフラッシュメモリをエミュレーションメモリとして使用することが可能となって、半導体プロセスの進歩でチップ内の動作速度が早くなっても、内蔵されているROMと同等の速度でアクセスすることができるようになり、エミュレーションメモリとして外部に高価な高速のメモリを外付けする必要がなくな

(15)

特開平10-214201

27

28

るばかりか、内蔵するROMのアクセスを禁止したり、エミュレーションメモリを外付けするために内部バスを外部に出力する機能や、デバッグ専用の割り込み入力を許可したり、ICEがマイコン内部の動作状態を知るためのモニタ信号を出力する機能などのデバッグ時専用の特殊なモードをマイコンに持たせる必要もなく、さらに、イベント検出によるブレーク機能をチップ内部に持っているため、あらかじめ設定されたイベントを検出すると、直ちにデバッグ割り込みを発生してユーザプログラムの実行を中断することができ、ブレークの発生に遅れがなくなり、また、マイコンをユーザ基板に直付けして、デバッグに必要な信号だけをコネクタを介してICEに接続することが可能となるため、ケーブルによる信号遅延の影響を受けることがなくなって、より最終製品の実装形態に近い状態でプログラムの開発/デバッグが行えるなどの効果がある。

【0104】請求項2記載の発明によれば、データを双方向にやりとりするデータ端子、やり取りされるデータの同期をとるためのクロックを双方向に送るクロック端子、およびデータ端子とクロック端子の信号の伝播方向を決める制御端子によって構成された専用入出力端子によって、内蔵されたデバッグ回路を外部のICEと接続するように構成したので、ICEとデバッグ回路間の信号線を最低本数に抑えることができる効果がある。

【0105】請求項3記載の発明によれば、書き換え不可能なROM上のプログラムにバグがあるとき、そのバグのあるプログラム部分のアドレスまで実行アドレスが進むとデバッグ回路よりデバッグ割り込みを発生させ、そのバグのあるプログラム部分を回避して、別用途された正しいプログラム部分に置き換えるように構成したので、製造完了後にROMに書き込んだプログラムにバグが見つかった場合でも、製品そのものが使用できなくなるといったリスクを回避できる効果がある。

【0106】請求項4記載の発明によれば、あらかじめ想定されていたアドレス範囲外にCPUのプログラム実行が及んだ場合に、デバッグ回路よりデバッグ割り込みを発生させてマイコンの最速検出を行うように構成したので、監視タイマを用いた場合のように、一定周期を経なくとも直ちに最速状態を検出することができ、より早期の最速検出、およびその最速からの復帰が可能となる効果がある。

【図面の簡単な説明】

【図1】 この発明の実施の形態1によるメモリ内蔵型のマイクロコンピュータの構成を示すブロック図である。

【図2】 この発明の実施の形態1におけるマイクロコンピュータのROMに格納するプログラムの開発環境を示すブロック図である。

【図3】 この発明の実施の形態1におけるマイクロコンピュータに内蔵されたデバッグ回路の構成を示すブ

ック図である。

【図4】 この発明の実施の形態1における各デバッグ制御レジスタのビット構成を示す説明図である。

【図5】 この発明の実施の形態1におけるデバッグ回路内の各レジスタのアドレス配置を示す説明図である。

【図6】 この発明の実施の形態1におけるCPUからの各デバッグ制御レジスタの読み出し/書き込みタイミングを示すタイミングチャートである。

【図7】 この発明の実施の形態1におけるICEからデバッグ回路内の各部レジスタを読み出すタイミングを示すタイミングチャートである。

【図8】 この発明の実施の形態1におけるレジスタ読み出し/書き込みコマンドのフォーマットを示す説明図である。

【図9】 この発明の実施の形態1におけるICEからデバッグ回路内の各レジスタへの書き込みタイミングを示すタイミングチャートである。

【図10】 この発明の実施の形態1におけるコードトレース時のトレース出力例を示すタイミングチャートである。

【図11】 この発明の実施の形態1におけるトレース属性のフォーマットを示す説明図である。

【図12】 この発明の実施の形態1におけるデータトレース時のデータ読み出し/書き込みの場合のトレース出力例を示すタイミングチャートである。

【図13】 この発明の実施の形態1におけるデータトレース時のデータ転送命令の場合のトレース出力例を示すタイミングチャートである。

【図14】 この発明の実施の形態1におけるマイクロコンピュータに内蔵されたフラッシュメモリの入出力信号を示す説明図である。

【図15】 この発明の実施の形態1におけるコマンド実行時のダウンロード制御回路の入出力タイミングを示すタイミングチャートである。

【図16】 この発明の実施の形態1におけるプログラムコマンド実行時のダウンロード制御回路の入出力タイミングを示すタイミングチャートである。

【図17】 この発明の実施の形態1におけるプログラムベリファイコマンド実行時のダウンロード制御回路の入出力タイミングを示すタイミングチャートである。

【図18】 この発明の実施の形態1におけるオートイレースコマンド実行時のダウンロード制御回路の入出力タイミングを示すタイミングチャートである。

【図19】 この発明の実施の形態1におけるデバッグモニタのアルゴリズムを示すフローチャートである。

【図20】 この発明の実施の形態2によるマイクロコンピュータのデバッグ回路のブレーク機能を用いたROMコレクションの処理を示すフローチャートである。

【図21】 従来のメモリ内蔵型のマイクロコンピュータの構成を示すブロック図である。

(16)

特開平10-214201

29

30

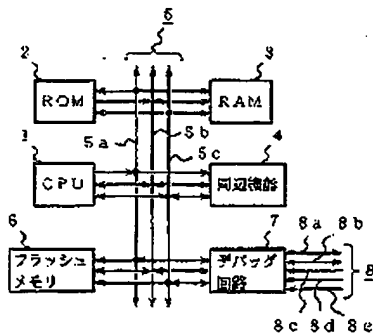
【図2】 従来のマイクロコンピュータのROMに格納するプログラムの開発環境を示すブロック図である。

【符号の説明】

1 CPU、2 ROM、5 内部バス、6 フラッシュメモリ、7 デバッグ回路、8 専用入出力端子、8*

* a GDバス（データ端子）、8b GCLK* 端子（クロック端子）、8c GOE* 端子（制御端子）、8d Vpp端子（制御端子）、8e RESET* 端子（制御端子）、11 マイコン、14 ICE。

【図1】

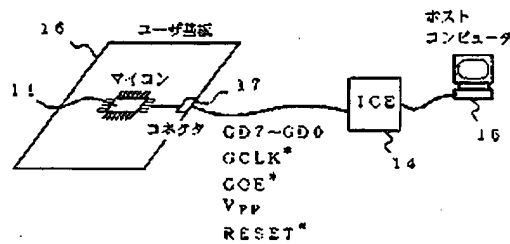


5: 内部バス
8: 専用入出力端子
8a: GDバス（データ端子）
8b: GCLK* 端子（クロック端子）
8c: GOE* 端子（制御端子）
8d: Vpp端子（制御端子）
8e: RESET* 端子（制御端子）

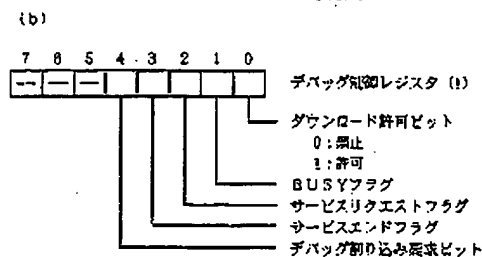
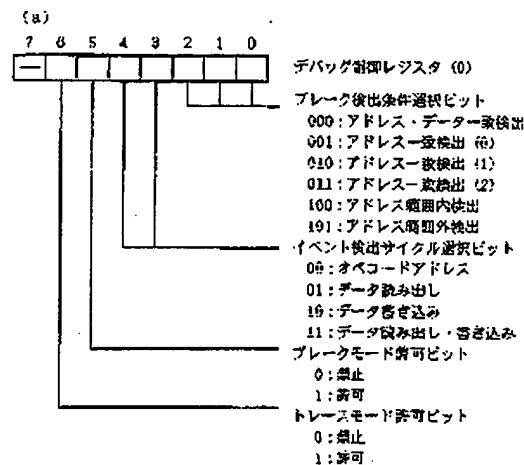
【図5】

番地	レジスタ名
66H	デバッグ制御レジスタ (0)
67H	デバッグ制御レジスタ (1)
68H	アドレス比較レジスタ (0) "L"
69H	アドレス比較レジスタ (0) "M"
6AH	アドレス比較レジスタ (0) "H"
6BH	アドレス比較レジスタ (1) "L"
6CH	アドレス比較レジスタ (1) "M"
6DH	アドレス比較レジスタ (1) "H"
6EH	データ比較レジスタ "L"
6FH	データ比較レジスタ "H"
70H	デバッグレジスタ (0)
71H	デバッグレジスタ (1)
72H	デバッグレジスタ (2)
73H	デバッグレジスタ (3)
74H	デバッグレジスタ (4)
75H	デバッグレジスタ (5)
76H	デバッグレジスタ (6)
77H	デバッグレジスタ (7)

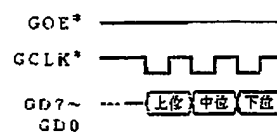
【図2】



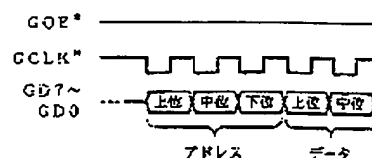
【図4】



【図10】



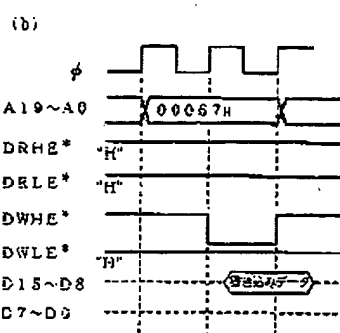
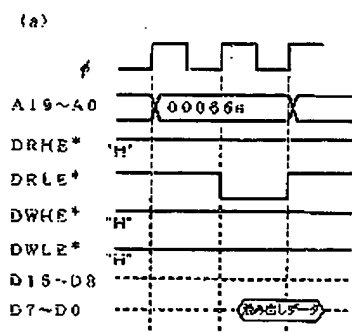
【図12】



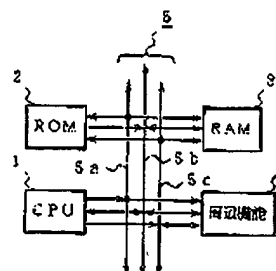
(18)

特開平10-214201

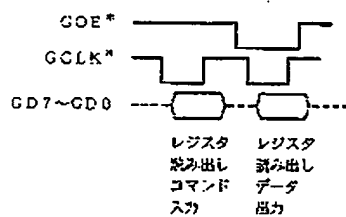
【図6】



【図21】



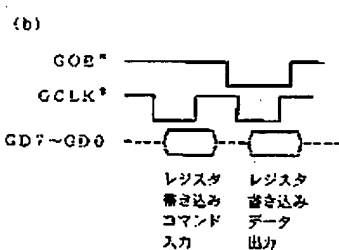
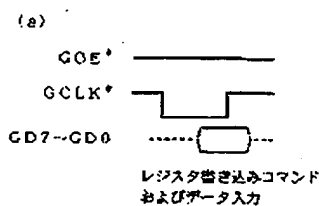
【図7】



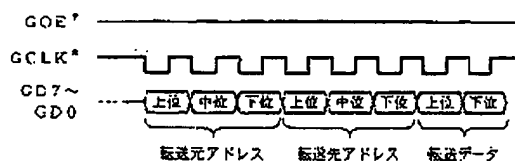
【図8】

ビット構成		7	6	5	4	3	2	1	0
レジスタ読み出し		0	0	1	68H~77Hの下位5ビット				
レジスタ書き込み	デバッグ制御レジスタ (0)	1	デバッグ制御レジスタ (0) 書き込みデータ						
	デバッグ制御レジスタ (1)	0	1	デバッグ制御レジスタ (1) 書き込みデータ					
	上記以外	0	0	0	68~77番地の下位5ビット				

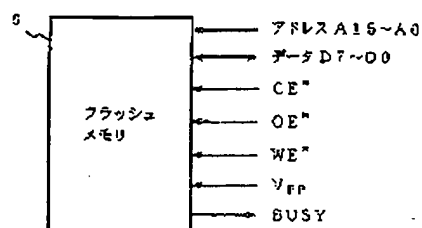
【図9】



【図13】



【図14】



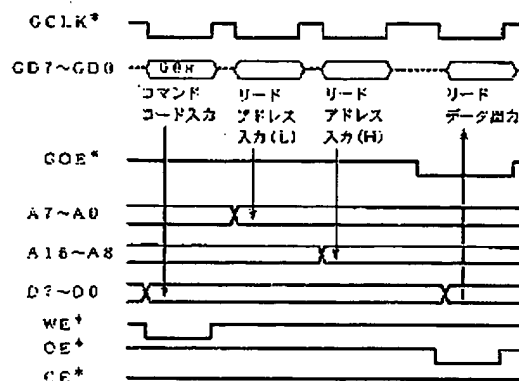
(19)

特開平10-214201

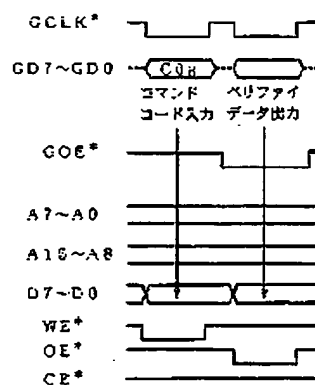
【図11】

		ビット			
		7	6	5	4
コード	分岐命令のオペコードアドレス			0	0
	分岐命令実行後のオペコードアドレス	0	0	0	1
	割り込み要求時のオペコードアドレス			1	0
	リセット解除後の飛び先アドレス			1	1
データ	リード			0	1
	偶数番地をバイトアクセス	0	1	1	0
	奇数番地をバイトアクセス			1	1
	偶数 / 奇数番地をワードアクセス			1	1
データ (データ転送 命令の場合)	ライト			0	1
	偶数番地をバイトアクセス	1	0	1	0
	奇数番地をバイトアクセス			1	1
	偶数 / 奇数番地をワードアクセス			1	1
GOE*端子入力を「L」から「H」にした時		0	1	0	0
ブレーク検出時		1	0	0	0

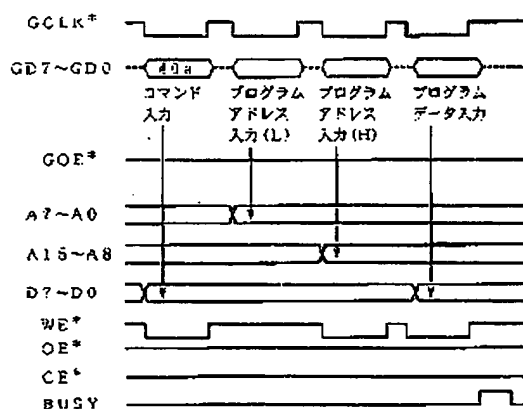
【図15】



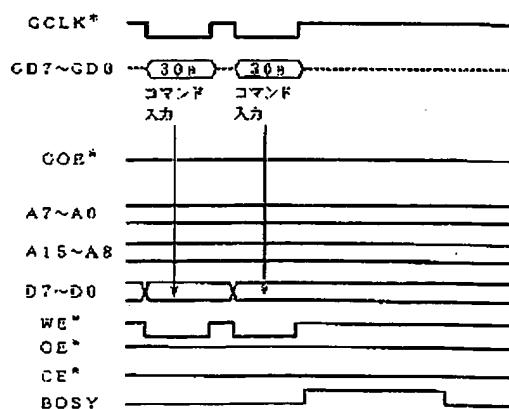
【図17】



【図16】



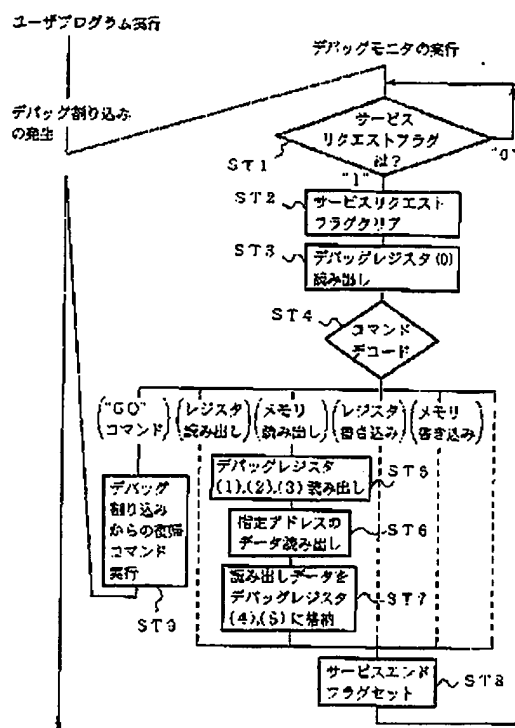
【図18】



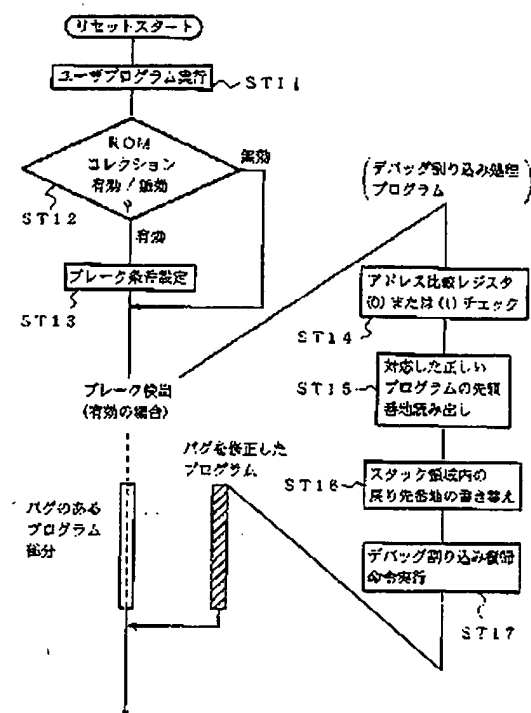
(20)

特開平10-214201

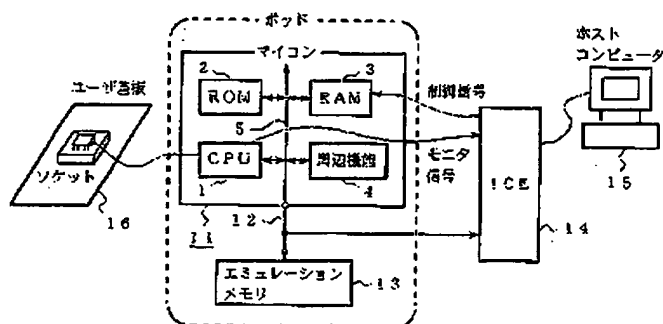
【図19】



【図20】



【図22】



(21)

特開平10-214201

フロントページの続き

(71)出願人 390009531
 インターナショナル・ビジネス・マシー
 ズ・コーポレイション
 INTERNATIONAL BUSIN
 ESS MASCHINES CORPO
 RATION
 アメリカ合衆国10504、ニューヨーク州
 アーモンク (香地なし)

(72)発明者 伊藤 栄
 東京都千代田区丸の内二丁目2番3号 三
 菱電機株式会社内

(72)発明者 神崎 照明
 東京都千代田区丸の内二丁目2番3号 三
 菱電機株式会社内

(72)発明者 赤月 忠之
 兵庫県伊丹市中央3丁目1番17号 三菱電
 機セミコンダクタソフトウェア株式会社内

(72)発明者 酒井 達也
 神奈川県藤沢市綱原町1番地 日本アイ・
 ビー・エム株式会社藤沢事業所内

(72)発明者 沼田 勉
 神奈川県藤沢市綱原町1番地 日本アイ・
 ビー・エム株式会社藤沢事業所内

(72)発明者 中村 泰博
 神奈川県藤沢市綱原町1番地 日本アイ・
 ビー・エム株式会社藤沢事業所内

? e pn=jp 10214201

Ref	Items	Index-term
E1	1	PN=JP 10214199
E2	1	PN=JP 10214200
E3	1	*PN=JP 10214201
E4	1	PN=JP 10214202
E5	1	PN=JP 10214203
E6	1	PN=JP 10214204
E7	1	PN=JP 10214205
E8	1	PN=JP 10214206
E9	1	PN=JP 10214207
E10	1	PN=JP 10214208
E11	1	PN=JP 10214209
E12	1	PN=JP 10214210

Enter P or PAGE for more

? s e3

S5 1 PN='JP 10214201'

? t s5/3,ab/all

5/3,AB/1

DIALOG(R)File 351:Derwent WPI

(c) 2004 Thomson Derwent. All rts. reserv.

012074234

WPI Acc No: 1998-491145/ 199842

XRPX Acc No: N98-384276

Microcomputer with built-in ROM - has EEPROM which is connected to CPU through internal bus and registers predetermined program

Patent Assignee: IBM CORP (IBM); MITSUBISHI DENKI SEMICONDUCTOR SOFTWARE (MITQ); MITSUBISHI ELECTRIC CORP (MITQ); INT BUSINESS MACHINES CORP (IBM); MITSUBISHI DENKI KK (MITQ); MITSUBISHI ELECTRIC SYSTEM LSI DESIGN CO (MITQ)

Inventor: AKATSUKI T; ITOH S; KANZAKI T; NAKAMURA Y; NUMATA T; SAKAI T

Number of Countries: 002 Number of Patents: 002

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
JP 10214201	A	19980811	JP 9715743	A	19970129	199842 B
US 6075941	A	20000613	US 9810538	A	19980122	200035

Priority Applications (No Type Date): JP 9715743 A 19970129

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
JP 10214201	A	21	G06F-011/28	
US 6075941	A		G06F-009/445	

Abstract (Basic): JP 10214201 A

The microcomputer has an EEPROM (6) in which a predetermined program is stored. The EEPROM is connected to a CPU (1) via an internal bus (5). A debugging circuit (7) is connected with an external emulator (14) through an exclusive input-output terminal (8). The debugging circuit communicates with the CPU via the internal bus.

The debugging circuit transmits information to the external emulator regarding the operating condition of the CPU. When operating condition of CPU corresponds to predetermined operating conditions, a debugging mode is set-up. During the debugging mode, a program code sent from the emulator is written into the EEPROM. The EEPROM contents are read out and are transmitted to the emulator.

ADVANTAGE - Offers minimum number of signal lines between debugging circuit and emulator. Enables EEPROM to be used as emulation memory. Generates debug interruption, immediately.

Dwg.1/22

? LOGOFF

1-1/1 Next page From 1 - 1 Count

Display format ----- Select the type of output. -----

Display checked documents

Check All

Uncheck All

☐ ** Result [P] ** Format(P807) 2004.12.10 1/ 1☐ C ☐ P

Application no/date: 1997- 15743[1997/ 1/29]
Date of request for examination: [2003/ 4/ 3]
Accelerated examination ()

Public disclosure no/date: 1998-214201  Translate [1998/ 8/11]

Examined publication no/date (old law): []

Registration no/date: []

Examined publication date (present law): []

PCT application no:

PCT publication no/date: []

Applicant:

Inventor: ITO SAKAE,KANZAKI TERUAKI,AKATSUKI TADAYUKI,SAKAI TATSUYA,NUMATA TSU
TOMU,NAKAMURA YASUHIRO

IPC: G06F 11/28 G06F 9/06 ,540 G06F 15/78 ,510

FI: G06F 11/28 L G06F 15/78 ,510K G06F 9/06 ,540L

F-Term: 5B042AA07,AA12,BB01,BB12,BB14,BB25,CC28,DD01,EA08,EA16,5B062AA08,CC03,
DD10,EE08,JJ03,JJ07,JJ08,5B076CA08,EA03,EB01,EC04,5B042GA07,HH01,HH11,HH25,H
H39,KK01,LA08,LA16

Expanded classification: 451,454

Fixed keyword: R131

Citation:

[19,2003.10. 3,] (,JP,Unexamined Patent Publication,1993159079)

[19,2003.10. 3,] (,JP,Unexamined Patent Publication,1996161191)

[19,2003.10. 3,] (,JP,Unexamined Patent Publication,1989232447)

[19,2003.10. 3,] (,JP,Unexamined Patent Publication,1996153018)

Title of invention: MICROCOMPUTER

Viability information of application: (

decision of rejection or appeal/trial decision of rejection)

Priority country/date/number: () [] ()

Domestic priority: [] ()

Original application number: ()

Original registration number: ()

Retroactive date:[]

No. of claims (2)

Classification of examiners decision/date: (decision of rejection) [2004/ 7/ 6]

Final examinational transaction/date: (

) []

Examination intermediate record:

(A63 1997/ 1/29,PATENT APPLICATIONUTILITY MODEL REGISTRATION APPLICATION, 2

1000:)(A523 1997/ 2/12,WRITTEN AMENDMENT, :)

(A111 1997/ 3/11,WRITTEN INVITATION OF AMENDMENT (APPLICATION), 21000:)

(A961 1997/ 4/ 1,CORRECTION DATA BY EX OFFICIO (FORMALITY), :)

(A961 1997/ 4/ 1,CORRECTION DATA BY EX OFFICIO (FORMALITY), :)

(A781 1997/ 4/ 7,WRITTEN STATEMENT, :)

(A961 1997/ 4/30,CORRECTION DATA BY EX OFFICIO (FORMALITY), :)

(A841 1997/10/30,WRITTEN REQUEST FOR CERTIFICATION OF RIGHT OF PRIORITY,

:)(A711 2003/ 3/ 4,NOTIFICATION OF CHANGE IN APPLICANT, 4200:)

(A7423 2003/ 3/ 4,NOTIFICATION OF APPOINTMENT OF POWER OF ATTORNEY, : A

)(A821 2003/ 3/ 4,WRITTEN SUPPLEMENT OF PROCEDURE, :)

(A7422 2003/ 4/ 3,NOTIFICATION OF ACCEPTANCE OF POWER OF ATTORNEY, : B)

(A621 2003/ 4/ 3,WRITTEN REQUEST FOR EXAMINATION, 92300:)

(A967 2003/ 4/16,RECOGNITION?ADDITION INFORMATION, :)

(A967 2003/ 4/16,RECOGNITION?ADDITION INFORMATION, :)

(A967 2003/ 4/16, RECOGNITION? ADDITION INFORMATION, :)
(A112 2003/ 4/22, WRITTEN INVITATION OF AMENDMENT (INTERMEDIATE DOCUMENT), : A)
(A071 2003/ 4/22, WRITTEN NOTICE (NOTICE OF RESUBMISSION), : B)
(A073 2003/ 6/ 3, DISPOSITION OF PROCEDURE DISMISSAL (AMENDMENT INVITATION), : A) (A9710072003/ 9/17, WRITTEN REPORT OF RETRIEVAL, :)
(A712 2003/10/ 1, NOTIFICATION OF CHANGE IN APPLICANT (GENERAL SUCCESSION), :)
(A9710112003/10/ 3, SITUATION LIST OF UTILIZATION OF OUTSIDE AGENCY FOR SEARCHING PATENT INFORMATION, :)
(A131 2003/10/ 7, WRITTEN NOTICE OF REASON FOR REJECTION, :)
(A53 2003/12/ 5, WRITTEN OPINION, :)
(A523 2003/12/ 5, WRITTEN AMENDMENT, :)
(A967 2003/12/11, RECOGNITION? ADDITION INFORMATION, :)
(A967 2003/12/11, RECOGNITION? ADDITION INFORMATION, :)
(A711 2004/ 5/27, NOTIFICATION OF CHANGE IN APPLICANT, 4200:)
(A821 2004/ 5/27, WRITTEN SUPPLEMENT OF PROCEDURE, :)
(A967 2004/ 6/29, RECOGNITION? ADDITION INFORMATION, :)
(A02 2004/ 7/ 6, DECISION OF REJECTION, :)
(A7422 2004/ 7/21, NOTIFICATION OF ACCEPTANCE OF POWER OF ATTORNEY, :)
(A821 2004/ 7/21, WRITTEN SUPPLEMENT OF PROCEDURE, :)
(A821 2004/ 7/21, WRITTEN SUPPLEMENT OF PROCEDURE, :)
(A821 2004/ 7/21, WRITTEN SUPPLEMENT OF PROCEDURE, :)
(A821 2004/ 7/21, WRITTEN SUPPLEMENT OF PROCEDURE, :)

*** Trial no/date [] Kind of trial [] ***

Demandant: -

Defendant: -

Opponent: -

Classification of trial decision of opposition/date: () []

Final disposition of trial or appeal/date: () []

Trial and opposition intermediate record:

Registration intermediate record:

Amount of annuity payment:

Extinction of right/Lapse date of right: () []

Proprietor: -

Status of register: ()

Check All

Uncheck All

Display checked documents

Display format ----- Select the type of output. -----

1-1/1

Next page

From 1 - 1 Count

Back to list